



montavista
VISION

2007

EMBEDDED LINUX DEVELOPERS CONFERENCE

Real Time Linux Technology

Paul E. McKenney

IBM Distinguished Engineer, Linux Technology Center



How I Got Here

Non-Real-Time Interlude

- Business-application programming
- ***Real-time programming (building control, security, acoustic navigation)***
- Systems administration (1986-8)
- Internet routing and congestion avoidance protocol (1988-1990)
- Parallel and NUMA algorithms, DYNIX/ptx, Digital Unix, AIX, Linux (1990-2004)
 - Some exposure to realtime via the MontaVista-lead PREEMPT effort interactions with RCU (2002-2004)
- ***Return to realtime:***
 - ***Parallel realtime algorithms in Linux (2004-present)***



Why Parallel Realtime?

Emergence of SMP Embedded Realtime Systems

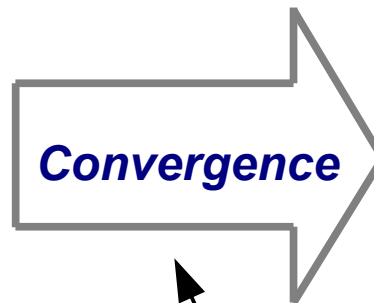
Traditional Systems

Traditional Realtime:
Few CPUs
Latency Guarantees
Non-Standard

OR

Traditional SMP:
Many CPUs
No Guarantees
Standard (and OSS)

But Not Both!!!



Emerging Systems

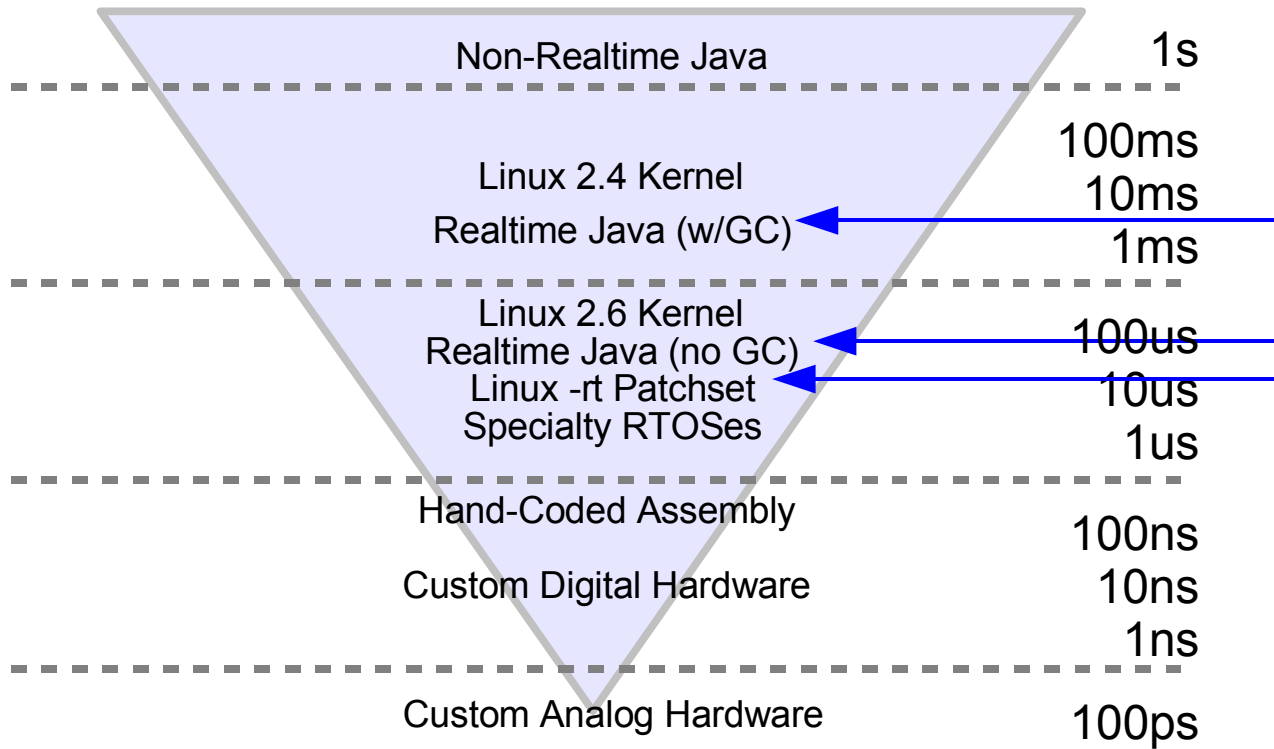
SMP Realtime:
Many CPUs
Latency Guarantees
Standard (and OSS)

- User Demand (DoD, Financial, Gaming, ...)
- Technological Changes Leading to Commodity SMP
 - Commodity Hardware Multithreading
 - Commodity Multi-Core Dies
 - Tens to Hundreds of CPUs per Die – Or More



Real-Time Regimes

Real-Time Regimes

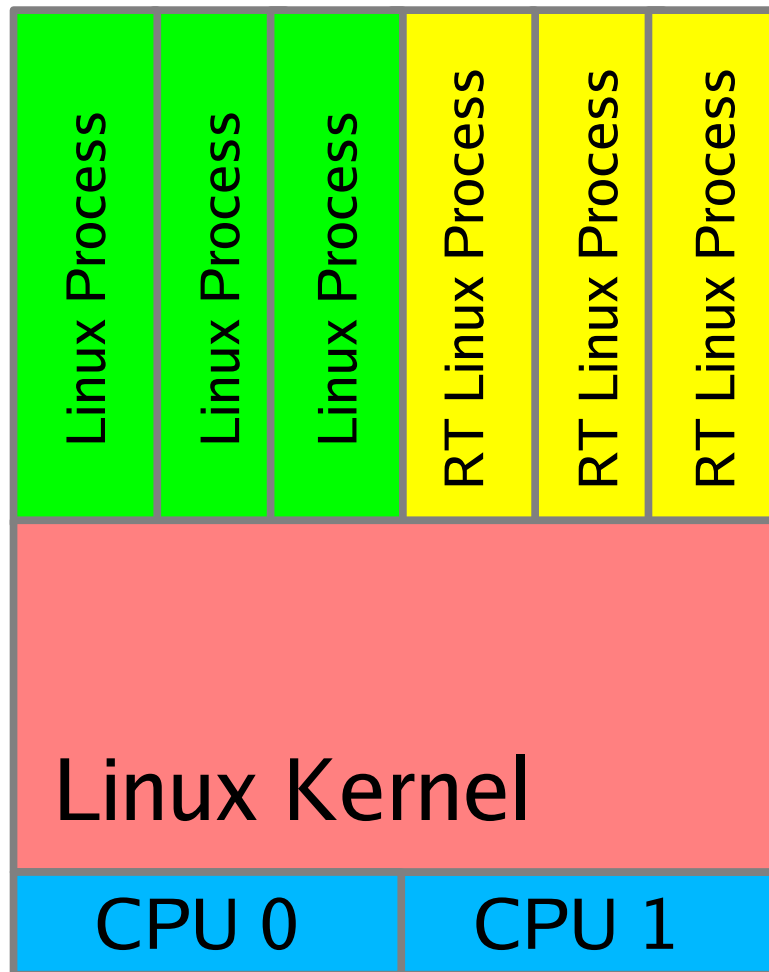




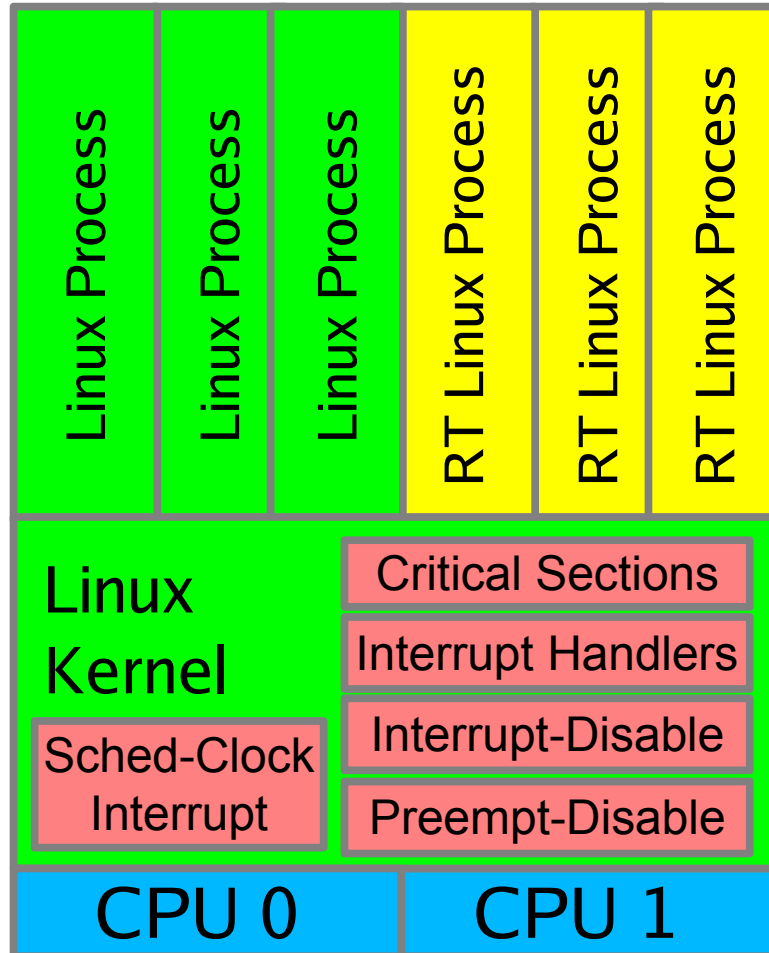
Preemption



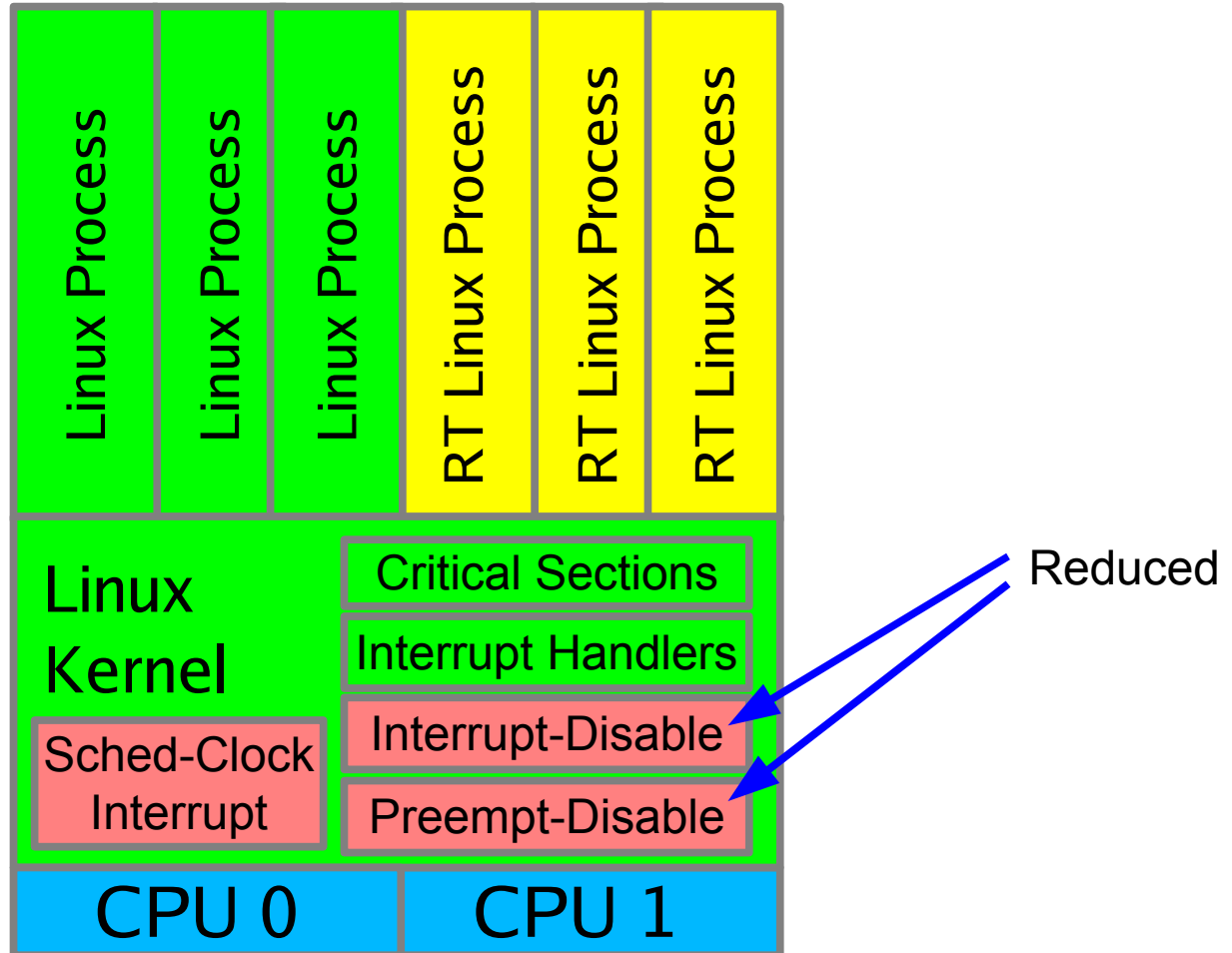
Vanilla Linux Kernel



Linux Kernel CONFIG_PREEMPT Build



Linux Kernel CONFIG_PREEMPT Build





Timers and -rt Patchset



Linux Timer Wheel at 1KHz





Linux Timer Wheel at 100KHz

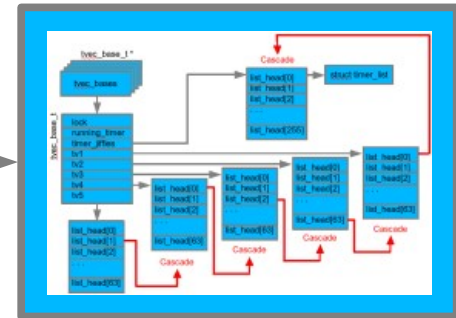


**Any
Questions?**

Solution: High-Resolution Timers

Timeouts: approximation OK, likely cancelled

add_timer(), mod_timer(), del_timer(), del_timer_sync(), ...



Timers: must be exact, rarely cancelled

hrtimer_init(), hrtimer_init_sleeper(), hrtimer_start(),
hrtimer_cancel(), hrtimer_forward(), ...





Priority Inversion and -rt Patchset

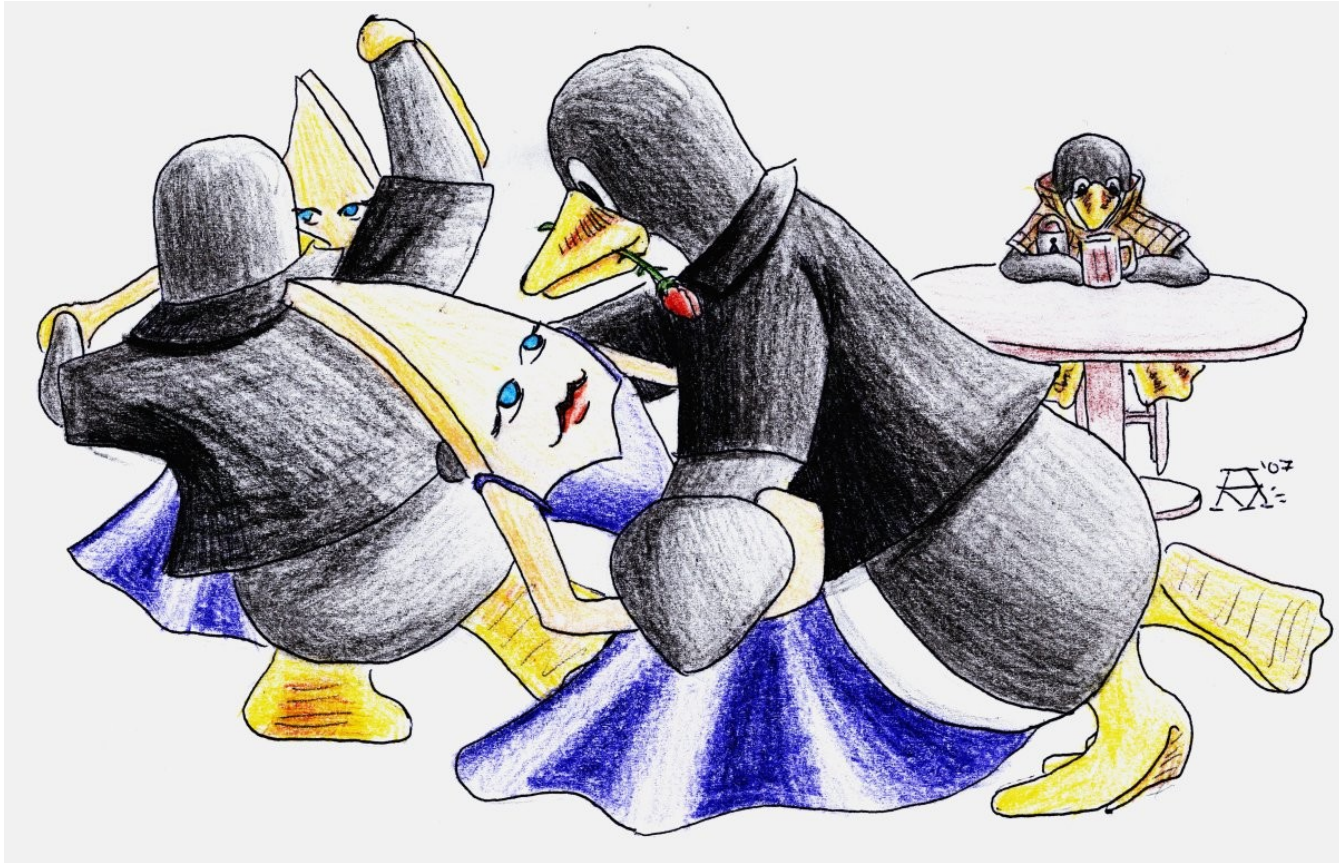
“Trapdoor” Metaphor for Priority Inheritance

- A dance floor...
 - CPUs dance with highest priority tasks (Tuxes)
- Warning: any attempt to apply this metaphor in reverse will probably not end well...

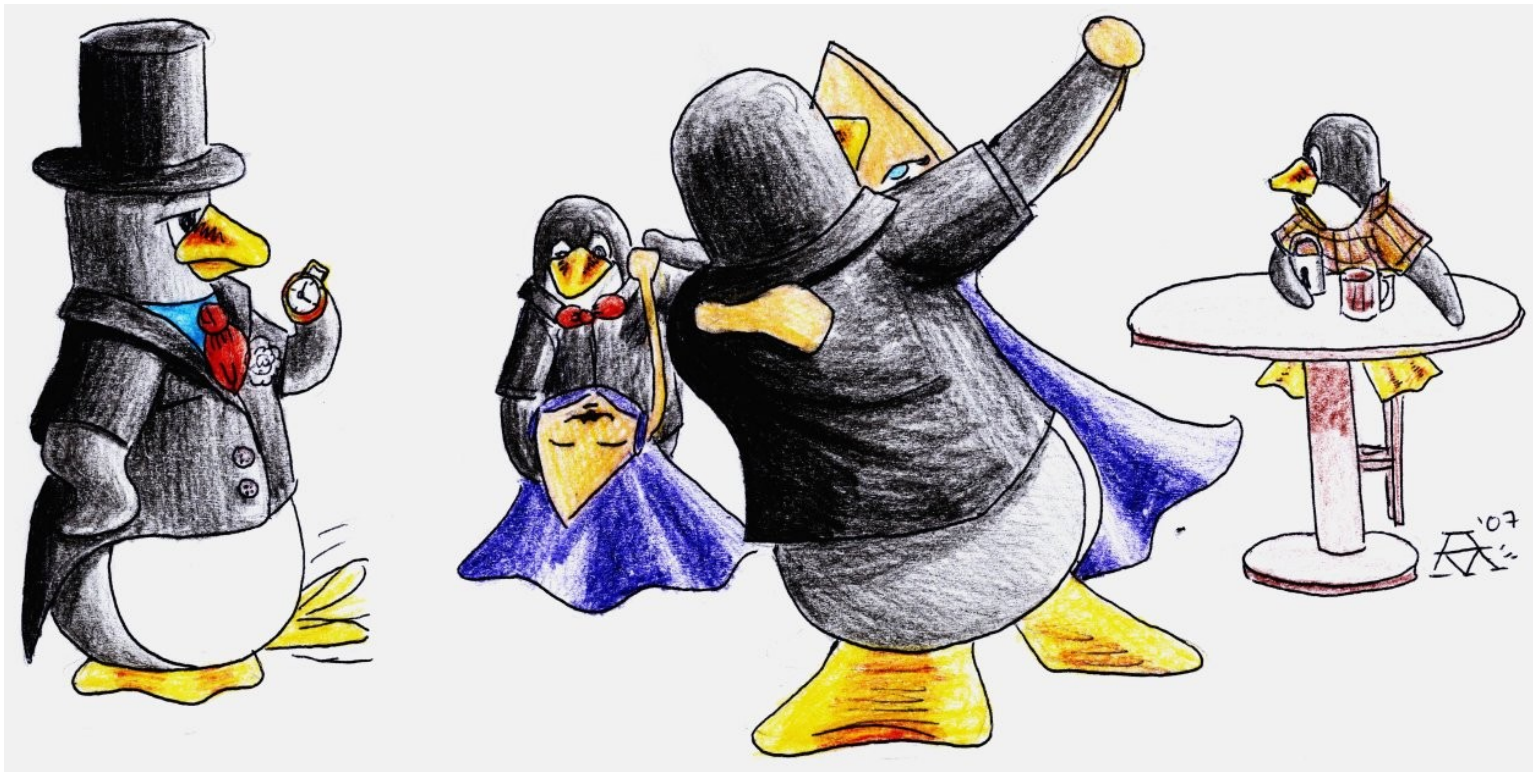
Priority Inheritance



Priority Inheritance



Priority Inheritance



Priority Inheritance



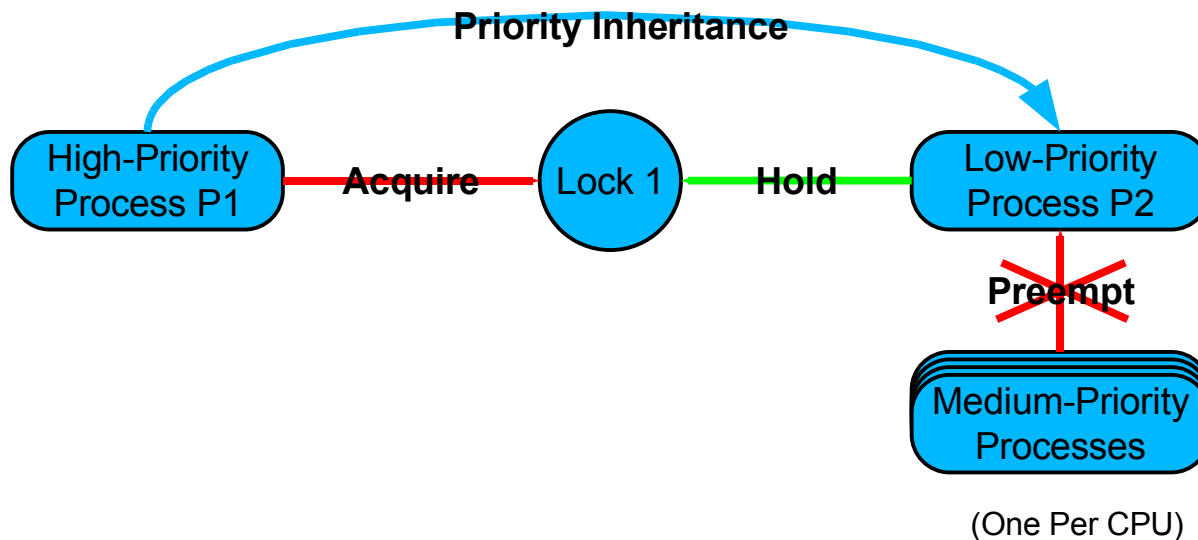


Priority Inheritance



Preventing Priority Inversion Outside the Dance Hall

- Trivial solution: Prohibit preemption while holding locks
 - But degrades latency!!! Especially for sleeplocks!!!!
- Simple solution: “Priority Inheritance”: P2 “inherits” P1's priority
 - But only while holding a lock that P1 is attempting to acquire
 - Standard solution, very heavily used
- Either way, prevent the low-priority process from being preempted





Priority Inheritance and Reader-Writer Locking



Priority Inheritance and Reader-Writer Locking

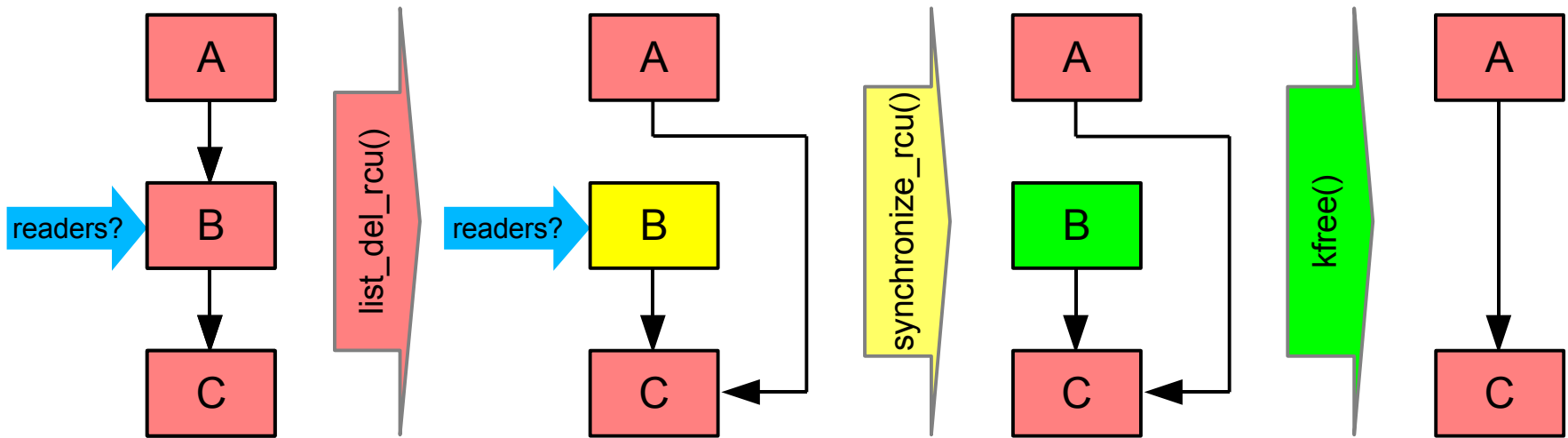




RCU

Example: RCU Removal From Linked List

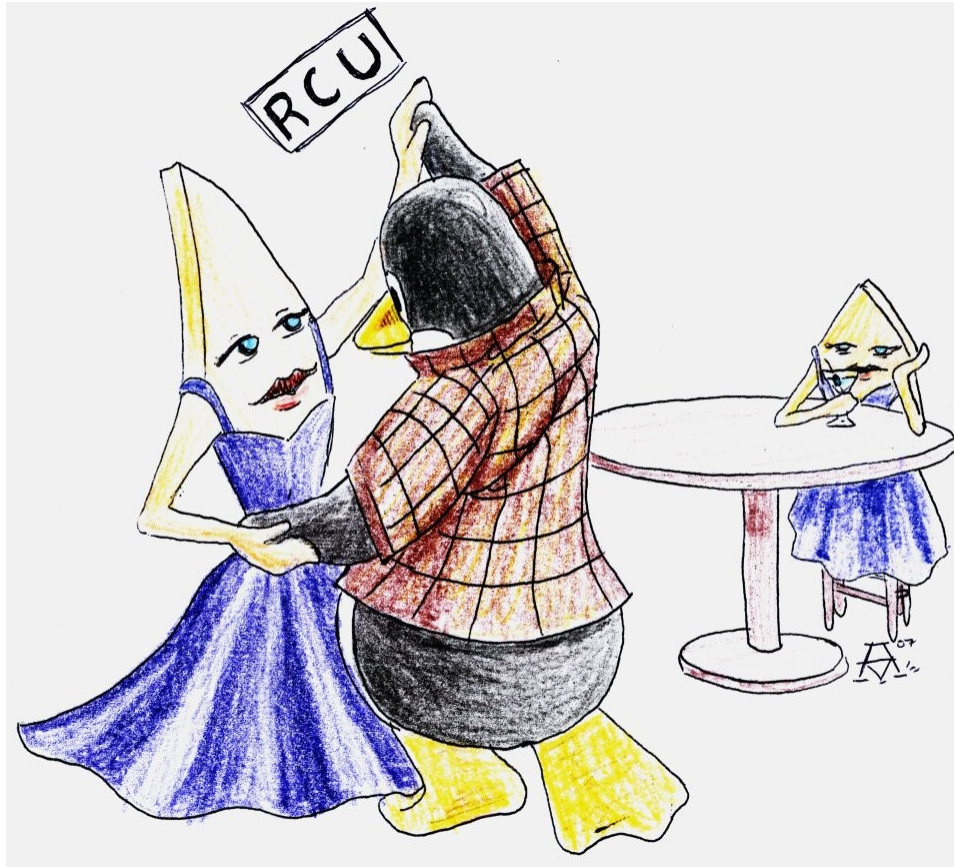
- Writer removes element B from the list (`list_del_rcu()`)
- Writer waits for all readers to finish (`synchronize_rcu()`)
- Writer can then free B (`kfree()`)



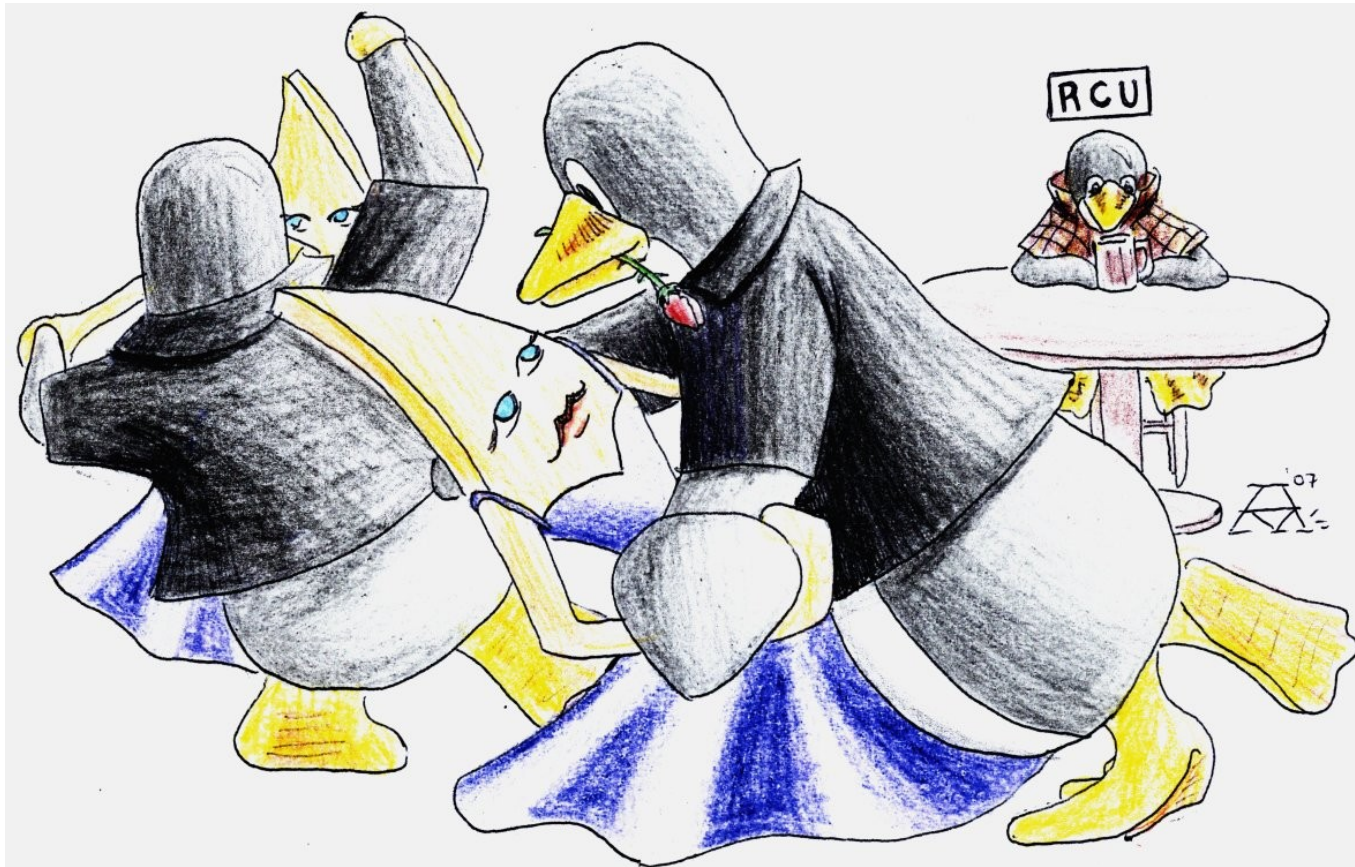
No more readers
referencing B!

Priority Inversion and RCU: Back to the Dance Hall

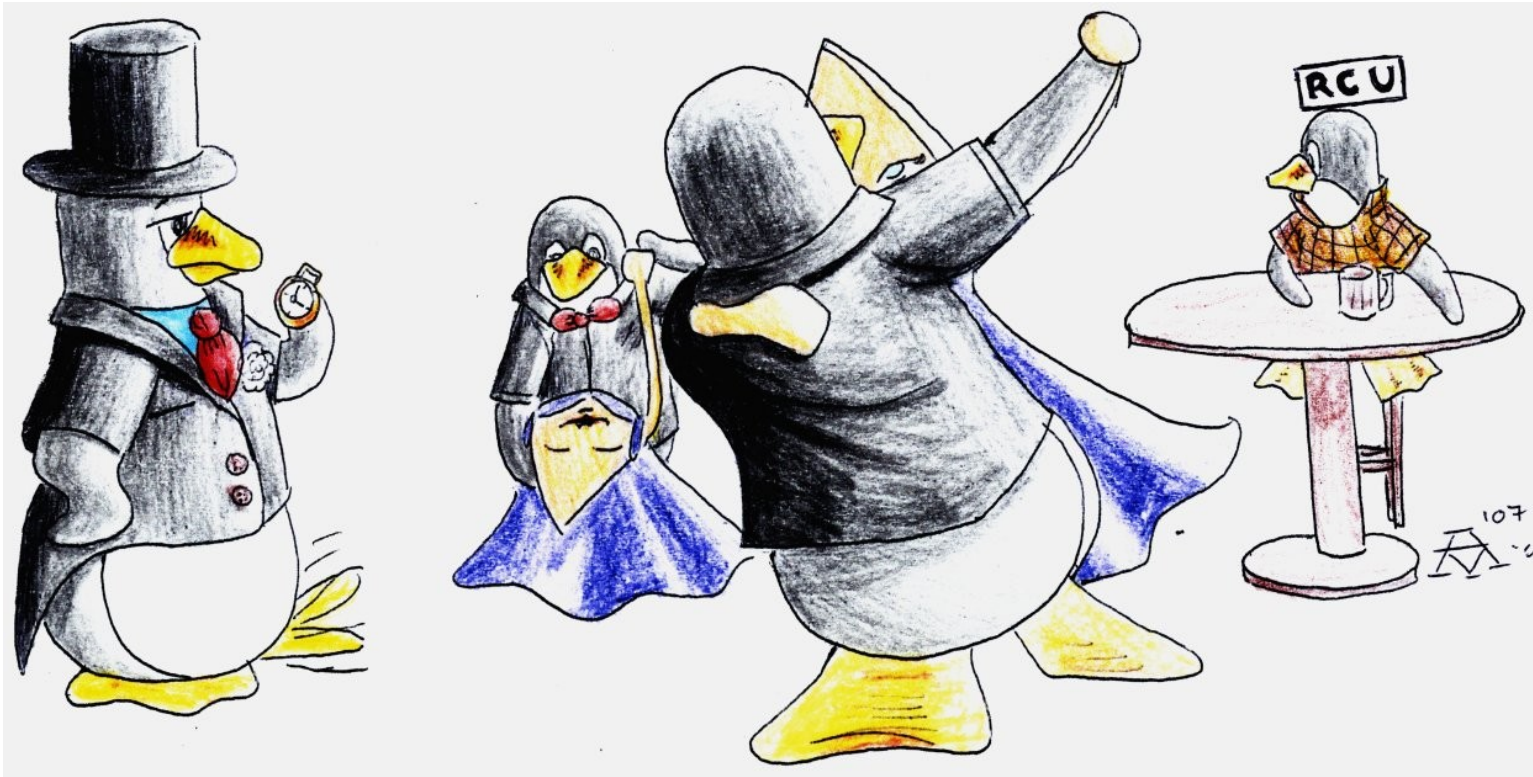
Priority Inversion and RCU



Priority Inversion and RCU



Priority Inversion and RCU



Priority Inversion and RCU

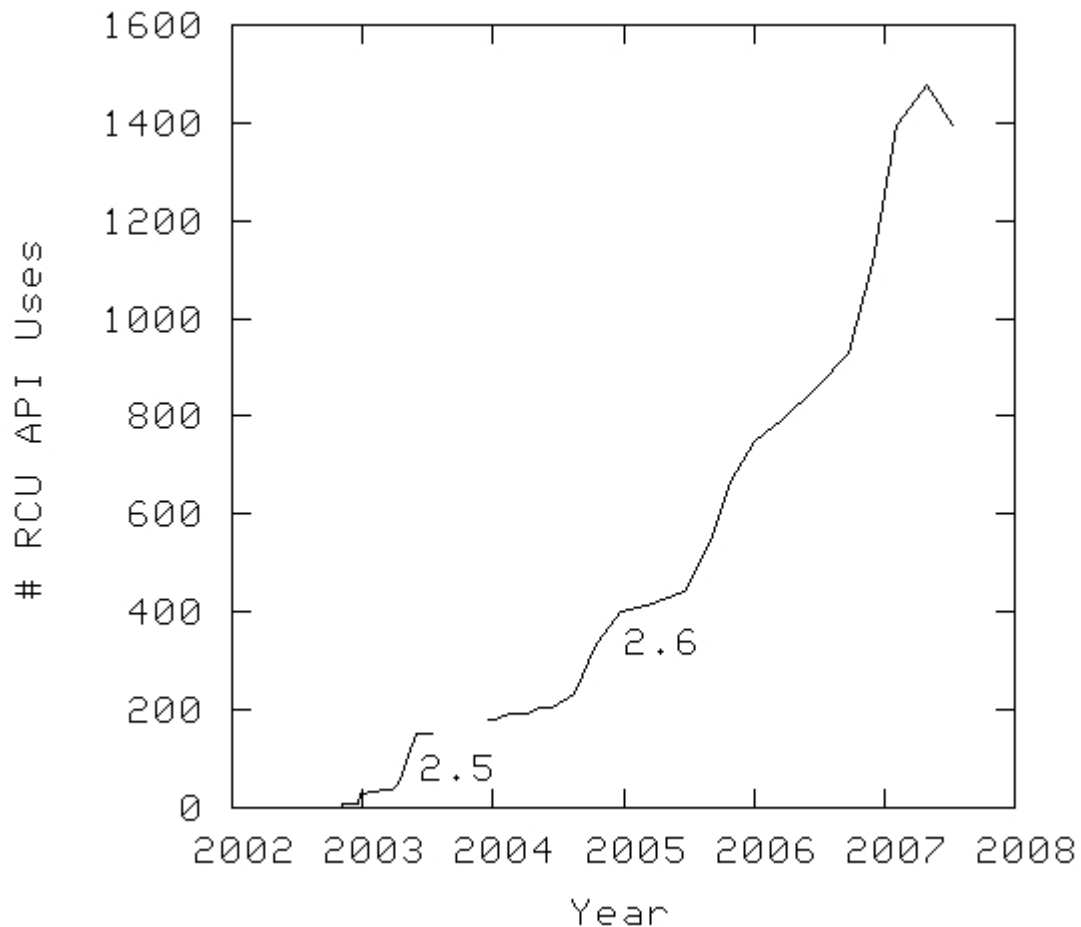




Can the Linux Community Handle RCU?

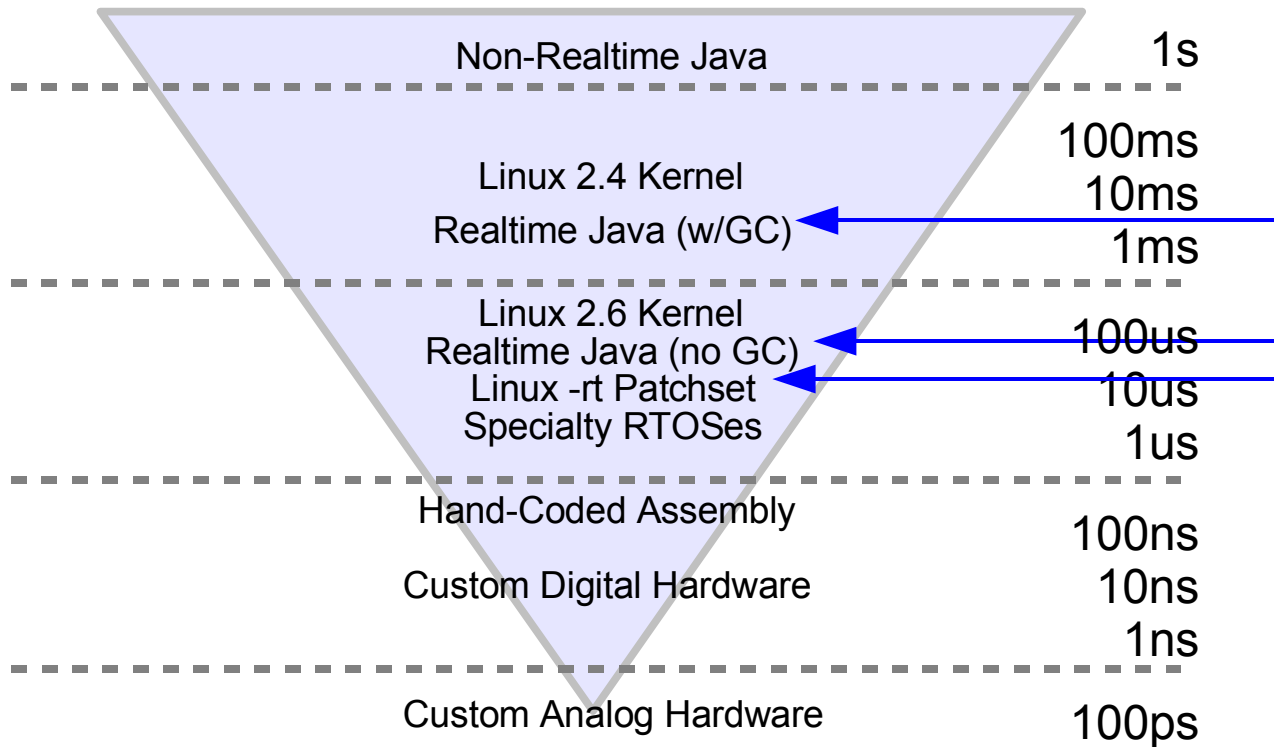


Linux Usage of RCU APIs



<http://www.rdrop.com/users/paulmck/RCU/linuxusage.html>

Summary: Realtime Regimes Redux





Summary

**Use
the right tool
for the job!!!**

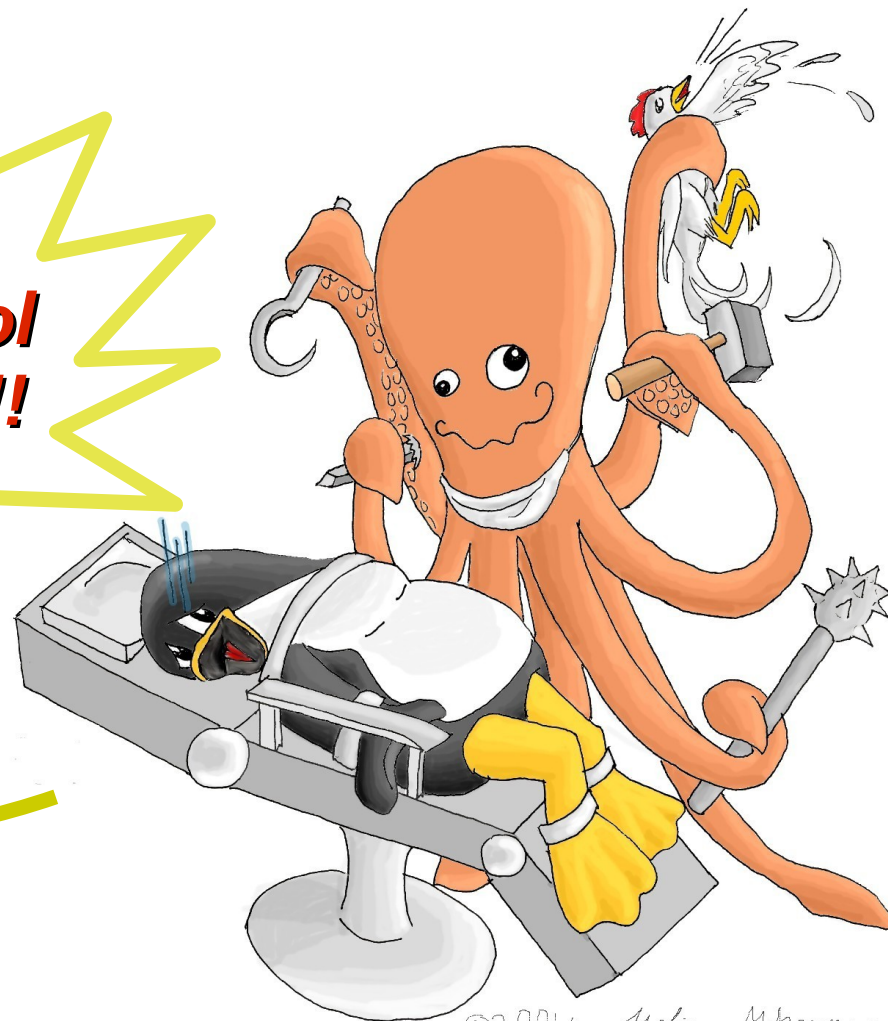


Image copyright © 2004 Melissa McKenney

©2004 Melissa McKenney



To Probe Deeper

- http://rt.wiki.kernel.org/index.php/Main_Page
- <http://people.redhat.com/mingo/realtime-preempt/>
 - But now: <http://www.kernel.org/pub/linux/kernel/projects/rt/>
- <http://www.linuxjournal.com/article/9361> (Linux Journal article)
- <http://www.ibm.com/common/ssi/fcgi-bin/ssialias?subtype=ca&infotype=an&appname=iSource&supplier=877&letternum=ENUSZP06-0365>
- <http://www.linutronix.de/>
- <http://www.mvista.com/products/realtime.html>
- Hollis Blanchard's "Virtualization – Not Just for Servers"
- My "Real Time Linux Technology: A Deeper Dive" (shameless plug)



"Controlling a laser with Linux is crazy, but everyone in this room is crazy in his own way. So if you want to use Linux to control an industrial welding laser, I have no problem with your using PREEMPT_RT." -- Linus Torvalds, July 2006

Legal Statement

This work represents the view of the author and does not necessarily represent the view of IBM.

IBM, IBM (logo), e-business (logo), pSeries, e (logo) server, and xSeries are trademarks or registered trademarks of International Business Machines Corporation in the United States and/or other countries.

Linux is a registered trademark of Linus Torvalds.

Other company, product, and service names may be trademarks or service marks of others.



Questions?